TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 03/24 pp325–342
DOI: 10.26599/TST.2023.9010031
Volume 29, Number 2, April 2024

Exploring a Promising Region and Enhancing Decision Space Diversity for Multimodal Multi-Objective Optimization

Fei Ming and Wenyin Gong*

Abstract: During the past decade, research efforts have been gradually directed to the widely existing yet less noticed multimodal multi-objective optimization problems (MMOPs) in the multi-objective optimization community. Recently, researchers have begun to investigate enhancing the decision space diversity and preserving valuable dominated solutions to overcome the shortage caused by a preference for objective space convergence. However, many existing methods still have limitations, such as giving unduly high priorities to convergence and insufficient ability to enhance decision space diversity. To overcome these shortcomings, this article aims to explore a promising region (PR) and enhance the decision space diversity for handling MMOPs. Unlike traditional methods, we propose the use of non-dominated solutions to determine a limited region in the PR in the decision space, where the Pareto sets (PSs) are included, and explore this region to assist in solving MMOPs. Furthermore, we develop a novel neighbor distance measure that is more suitable for the complex geometry of PSs in the decision space than the crowding distance. Based on the above methods, we propose a novel dual-population-based coevolutionary algorithm. Experimental studies on three benchmark test suites demonstrates that our proposed methods can achieve promising performance and versatility on different MMOPs. The effectiveness of the proposed neighbor distance has also been justified through comparisons with crowding distance methods.

Key words: multimodal multi-objective optimization; evolutionary algorithms; promising region; neighbor distance; decision space; coevolution

1 Introduction

Multimodal multi-objective optimization problem (MMOP) is a type of multi-objective optimization problem that contains more than one Pareto set (PS) in the decision space corresponding to the same Pareto front (PF) in the objective space. It is widely existent in scientific research and real-world applications such as credit card fraud detection[1], multi-objective knapsack problem[2], neural architecture search[3], furnace-grouping problem[4], home health care services[5], and manufacturing scheduling[6]. In the past decade, solving MMOPs attracted increasing research attention in the multi-objective optimization community[7, 8].

Generally, an MMOP can be formulated as follows:

$$\begin{aligned} &\text{Minimize} \quad F(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_m(\boldsymbol{x}))^{\mathrm{T}}, \\ &\text{subject to} \quad \boldsymbol{x} \in \boldsymbol{S} \end{aligned} \tag{1}$$

where m represents the number of objective functions, $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$ represents an n-dimensional decision vector, n represents the number of decision variables, and $\boldsymbol{x} \in \boldsymbol{S}$ and $\boldsymbol{S} \subseteq \boldsymbol{S}^n$ represent the search spaces†, respectively.

• Fei Ming and Wenyin Gong are with School of Computer Science, China University of Geosciences, Wuhan 430074, China. E-mail: {feiming; wygong}@cug.edu.cn.
∗ To whom correspondence should be addressed.
Manuscript received: 2023-02-19; revised: 2023-03-06; accepted: 2023-04-06

† Basic concepts of MMOPs can be found in Section S2.2 in the Electronic Supplementary Material (ESM).

Figure 1 depicts an illustration of MMOP features. The left figure shows two solutions, A and B, which are marked in black and gray dots, on the PF in the objective space. The right figure shows three PSs in the decision space corresponding to the PF and the Pareto optimal solutions corresponding to A and B. Both A and B have three corresponding Pareto optimal solutions on each of the three PSs.

Given that the goal of solving an MMOP is to provide as many Pareto optimal solutions (i.e., PSs) as possible for the decision makers to select[9], enhancing the decision space diversity on the PSs aside from the objective space convergence and diversity is a crucial issue. Recent research works focused on balancing objective space convergence, objective space diversity, and decision space diversity[10] to better solve MMOPs. Particularly, some tended to preserve solutions considering not only the objective space diversity but also the decision space diversity[9, 11], while others used auxiliary techniques (i.e., clustering[12], archive[13], and reference vectors[14]) to maintain the diversity in the decision space to assist in the search for the PF and PSs. However, most, if not all, existing multimodal multi-objective optimization evolutionary algorithms (MMOEAs) still suffer from the following two limitations:

First, most existing MMOEAs have an unduly high priority for convergence, eliminating some valuable well-spread dominated solutions in the decision space that can help explore and exploit the PSs[14]. In other words, dominated solutions are rarely considered to be preserved in the evolutionary process.

Second, the widely used crowding distance[15] is designed to enhance the diversity of the PF in the objective space. However, it is ineffective in some cases for the PSs in the decision space because the geometric characteristic of PSs varies from that of PF‡.

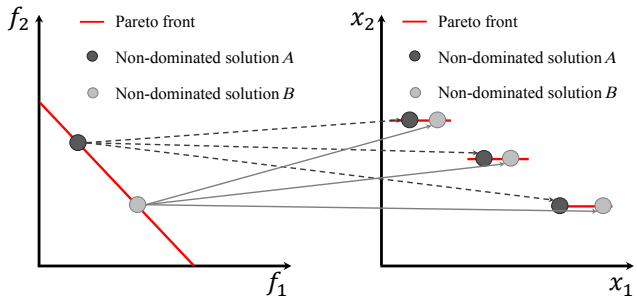


Fig. 1 Illustration of MMOP features. The left figure shows two solutions (A and B) on the PF in the objective space, while the right figure shows three PSs corresponding to the PF and the corresponding Pareto optimal solutions to A and B.

To overcome the above limitations, this paper proposes to explore a promising region (PR) in the decision space using a neighbor distance diversity metric for handling MMOPs. The main contributions of this work can be summarized as follows.

(1) We propose a concept in MMOPs named PR, which is defined by the obtained non-dominated solutions. It is a subspace of the decision space where the PSs can be partially or fully included. Moreover, we propose an η threshold technique to further restrict smaller areas named limited region (LR) in the PR that includes PSs. Furthermore, exploring the LR evenly (i.e., maintaining a good distribution in the LR and ignoring objective space convergence and objective space diversity) can facilitate the exploration and exploitation of PSs.

(2) We propose a novel diversity metric named neighbor distance, which utilizes the sum of Euclidean distances of the solution to a set of its neighbor solutions, rather than only two adjacent solutions in each dimension in the crowding distance, to measure its sparsity in the decision space. The neighbor distance can favor sparse solutions in dealing with complex (e.g., non-functional, piecewise, symmetrical, and discontinuous) geometric shapes of PSs.

(3) Based on the above-mentioned techniques, we propose a novel dual-population-based coevolutionary MMOEA. A convergence-prior-guided population can maintain a non-dominated solution set with an outstanding degree of diversity in the decision space as the final output, while a PR-guided population can explore the decision space to assist the search for PSs by maintaining a well-distributed solution set in the LR.

Experimental studies on three benchmark test suites of 40 instances verifies that our proposed methods can achieve considerably promising performance and versatility on different MMOPs, indicating that exploring the PR and LR can facilitate the search for PSs. Additionally, we verifies that in dealing with MMOPs, our proposed neighbor distance performs better in enhancing the decision space diversity than three crowding distance metrics[9–11].

The remainder of this article is organized as follows. Section 2 briefly reviews the existing MMOEAs and presents the motivations of this work. Section 3 elaborates on our proposed methods, including the PR and LR concepts, the neighbor distance, and

‡ Detalied analysis are presented in Section 2.2 and Section 3.2.

the proposed MMOPR algorithm. Furthermore, the experimental studies are detailed in Section 4. Finally, the conclusions and future work directions are presented in Section 5.

2 Existing MMOEA and Motivation

2.1 Existing MMOEAs

Existing MMOEAs can be roughly categorized into two classes based on whether objective space convergence is preferred or balanced.

2.1.1 Convergence-prior MMOEAs

The first category of MMOEAs prefers objective space convergence (i.e., the approximation to the PF) during the evolutionary process. Liang et al.[11] proposed the DN-NSGA-II algorithm that considered crowding distance in the objective and decision spaces for mating and environmental selections among non-dominated solutions. Yue et al.[9] proposed a particle swarm optimizer (PSO) that identifies the niches by a ring topology and enhanced the decision space diversity among non-dominated solutions by a special crowding distance. Tanabe and Ishibuchi[16] introduced a decomposition-based framework that associated each non-dominated solution to a reference vector, and then compared the solutions in the same subproblem based on their decision space diversity. Liu et al.[13] proposed the use of two archives to maintain the diversities in objective and decision spaces, and then tailored a recombination strategy that detects convergence-related variables and generates final PSs using the values of these variables. Lin et al.[12] proposed a dual clustering method that first uses clustering in the decision space and then employs clustering in the objective space to maintain the diversity of non-dominated solutions. Peng and Ishibuchi[17] proposed the use of a subset selection method to select a subset from the non-dominated solutions with an outstanding degree of diversity in both spaces. Fan and Yan[18] proposed a zoning search method to divide the entire search space into various subspaces and then reduce the size of the search space and the complexity of the problem to better promote decision space diversity among non-dominated solutions. Li et al.[19] proposed the use of a self-organizing quantum-inspired PSO that determines the neighbor leader of particles and adopts a zone-searching method to update particles. Further, they proposed a special archive to maintain Pareto optimal solutions. Li et al.[20] proposed a two-archive strategy that uses one archive for maintaining objective space diversity and another for decision space diversity among non-dominated solutions. Liang et al.[21] proposed a clustering-based crowding distance to calculate the diversity metric to maintain the diversity of non-dominated solutions. Qu et al.[22] proposed dividing the decision space into grids and maintaining the decision space diversity of non-dominated solutions through grid-based density.

2.1.2 Convergence-balanced MMOEAs

The second category of MMOEAs aims to balance objective space and decision space diversity, and tries to overcome the shortage of convergence-prior methods. Compared with the first category, this is an emerging topic in the field of multimodal multi-objective optimization. Li et al.[14] proposed a weighted indicator that considered decision space diversity when evaluating the convergence performance of solutions. They also maintained an archive with the aim of approximating the PF. Zhang et al.[23] proposed a two-stage double-niched evolution strategy that uses a niching in the decision space in the first stage and a niching in both spaces in the second stage. They also proposed a decision density self-adaptive strategy to balance the diversities in both spaces. Han et al.[24] proposed an information-utilization method that randomly extracts decision variable information from the current optimal solutions to construct an information vector to generate elite solutions for MMOEAs. Given that much decision variable information is used, this method can avoid quickly converging to and being trapped in easy-to-find PSs. Liu et al.[25] proposed to consider the local convergence quality of solutions to evaluate the decision space distances and estimated the decision space density values based on the transformed distances. Yue et al.[26] proposed an improved crowding distance that considers the objective and decision space diversities, in which if a dominated solution has a relatively larger crowding distance, it has a higher chance to survive.

2.2 Motivations

Among the above-mentioned methods, the first category (i.e., convergence-prior) of MMOEAs always prefers objective space convergence. These methods discard all dominated solutions during the evolutionary process. Nevertheless, dominated solutions with a good degree of decision space diversity might be valuable in exploring and exploiting PSs[14, 23, 25]. However, these methods suffer from the limitation of assigning an absolute priority to convergence. For the second category, there

are currently few studies that focus on preserving promising dominated solutions. Although convergence and diversity are balanced in these methods, convergence still has a high priority (e.g., convergence is taken into consideration either in evaluating solutions or during the selection procedure). As pointed out in Ref. [14], there is currently no work that considers solutions of worse objective values (i.e., dominated solutions). Inspired by the concept of constraint-ignorance[27] and promising area[28] in the constrained multi-objective optimization field, we come up with the idea of maintaining a well-spread solution set in a potential region where PSs are included to assist the exploration and exploitation of PSs. Thus, to fill the above-mentioned research gap and better balance convergence and diversities in both spaces, we propose a novel concept named PR.

We can find that many of the above-reviewed MMOEAs use the original or improved crowding distance[15] as a metric of decision space diversity. The crowding distance has been originally proposed for enhancing the objective space diversity (i.e., diversity on the PF) and has become considerably popular owing to its effectiveness in practice. It first calculates the distance from the solution to its two adjacent solutions in each objective and sums the distances on all objectives. In the object space, given that the objects conflict with each other, the shape of the PF must be a piecewise continuous manifold that is subject to a functional relation[29]. However, because the decision variables do not conflict with each other, the shape of PSs can be typically more complex and nonfunctional. Thus, the crowding distance is less effective in enhancing the decision space diversity on the PSs.

To better illustrate this, Fig. 2 depicts an artificial scene in a two-dimensional decision space. As shown in Fig. 2a, there are three PSs and eight solutions. We assume that four solutions must be selected to survive. Subsequently, we depict the crowding distance value of solutions A, B, a, and b. Given that the value of x_2 are equal on each PS, the crowding distances are represented by the gray lines (for a and b) and black lines (for A and B). Apparently, solutions a and b have larger crowding distances than solutions A and B. However, ideally A and B should also be selected because they are distributed on sparser PSs and can facilitate better exploitation of the uppermost two PSs. Furthermore, solutions A and B (marked in black dots) have better distance values than a and b if we generate a neighbor area of each solution and evaluate the diversity of these

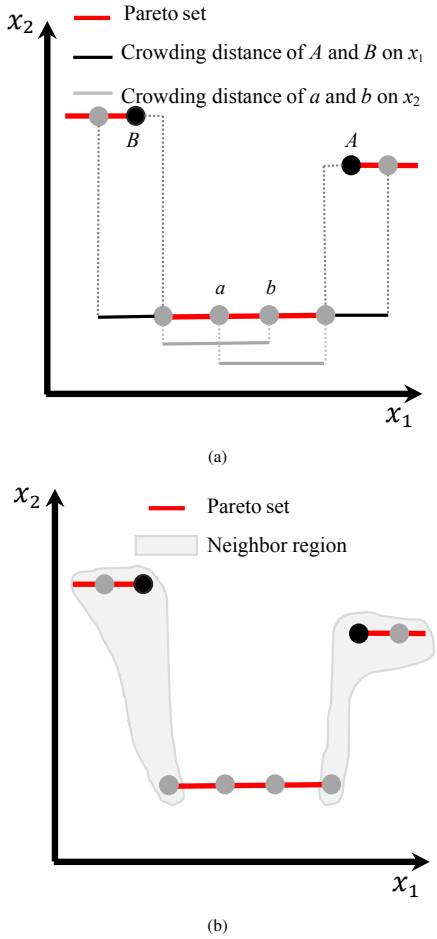


Fig. 2 Artificial scene in a 2D decision space. (a) Illustration of the shortages of the crowding distance and (b) the proposed neighbor concept.

solutions using the information of all neighbor solutions, as shown in Fig. 2b.

In the next section, we present in detail our proposed methods, including the PR and LR concepts, the neighbor distance, and the proposed MMOPR algorithm.

3 Proposed Method

3.1 Promising region and limited region

3.1.1 Promising region

We elaborate on the proposed PR and LR concepts in the first part of this section. The definition of identifying a solution in the PR can be expressed as follows:

Definition 3.1 Given a non-dominated solution set $\mathcal{S}, z^{\max} = \max x_i, \boldsymbol{x} \in \mathcal{S}, i = 1, 2, \ldots, n$. A solution $\boldsymbol{y}$ is in the PR if $\exists y_i \leqslant z_i^{\max}, i = 1, 2, \ldots, n$.

In other words, if the value of any dimension (decision variable) of a solution is smaller than the maximum value of the non-dominated solution set in this dimension, it is in the PR.

We present two ideal and two general circumstances in Figs. 3 and 4, respectively, to better demonstrate the rationale behind the PR concept. The maximum values on the two dimensions (x_1 and x_2) are depicted in blue dashed lines, and the PR is surrounded by blue dashed lines. Figure 3 presents two ideal circumstances in which the PSs are fully included in the PR. In these two situations, the maximum decision variable values of the obtained solution set define a PR where all the PSs are included. Thus, exploring the PR not only helps explore the undetected PSs (Fig. 3a) but also exploits the sparse or undetected areas on the PSs (Fig. 3b).

Figure 4 depicts two general circumstances in which using the maximum values on the two dimensions is insufficient to fully cover the PSs. In Fig. 4a, the undetected area surrounded by the green dashed lines is not covered by the maximum x_2. However, given that the maximum x_1 reaches the right side of the PSs, according to Definition 3.1, the PR is surrounded by the maximum x_1 and the upper boundary of x_2 because all the solutions (represented by $\boldsymbol{x}^s$) satisfy $x_1^s \leqslant x_1^{\max}$. Consequently, the PSs can be fully included in the PR. Figure 4b presents a more general circumstance where the PSs can be only partially included in the PR and the upper right segment of a PS is outside the PR. Nevertheless, in this situation, exploring the PR can help find undetected areas on the PSs. The PSs can be gradually detected and covered if the newly generated solutions in these undetected areas in the PR can be detected, i.e., the decision space diversity can be enhanced.

In summary, the PR can partially or fully include the PSs, and exploring the PR can facilitate a better search

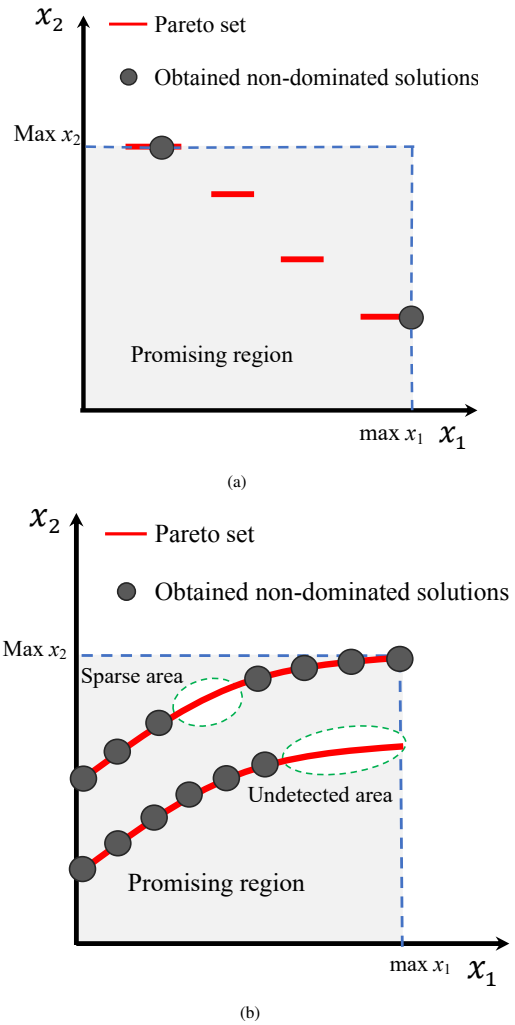


Fig. 3 Illustration of the proposed promising region concept under ideal circumstances. The promising region, where all the PS segments are included, is surrounded by blue dashed lines.

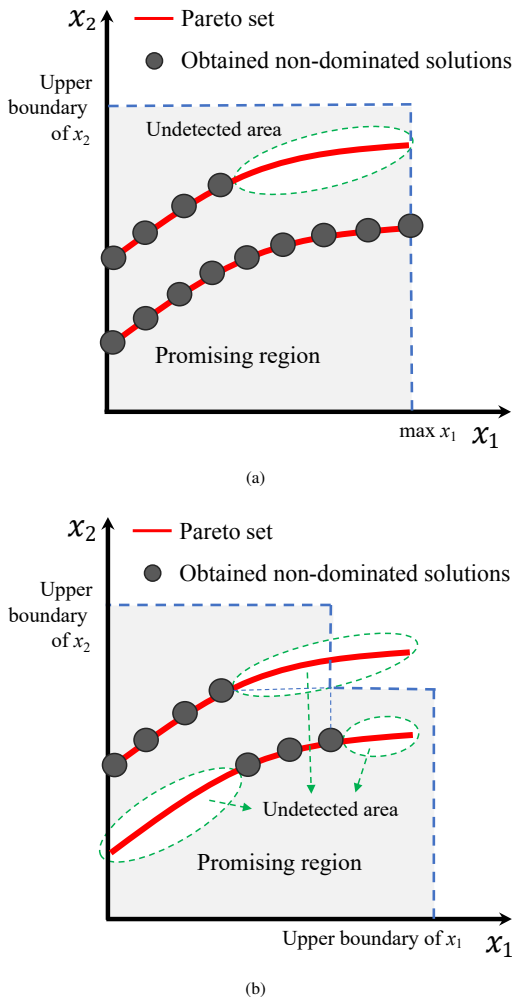


Fig. 4 Illustration of the proposed promising region concept under general circumstances. The promising region includes all the PS segments (a) and can promote the search for undetected areas (b).

for the PSs.

3.1.2 Limited region

Although the PR has the above-mentioned outstanding properties, the PR can be considerably large if the decision space is large. Consequently, it will be considerably challenging to locate the PSs when exploring a large PR. Thus, with the aim of reducing the search space and facilitating the location of PSs, we propose an η threshold technique to further restrict an LR in the PR.

First, we normalize the values of the objective functions to eliminate the influence of different scales. The maximum ($z^{\max}$) and minimum ($z^{\min}$) values are obtained from all solutions of the union of the solution set $\mathcal{S}$ and the non-dominated solution set $\mathcal{N}$ as follows:

$$z_i^{\max} = \max\ f_i, \boldsymbol{x} \in \mathcal{S} \cup \mathcal{N}, i = 1, 2, \ldots, m \tag{2}$$

and

$$z_i^{\min} = \min\ f_i, \boldsymbol{x} \in \mathcal{S} \cup \mathcal{N}, i = 1, 2, \ldots, m \tag{3}$$

Then the normalization is performing using

$$\bar{f}_i(\boldsymbol{x}) = \frac{f_i(\boldsymbol{x}) - z_i^{\min}}{z_i^{\max} - z_i^{\min}} \tag{4}$$

Subsequently, we calculate the average sum of objective function values of the obtained non-dominated solution set as follows:

$$\text{obj}^{\text{average}} = \frac{\sum_{\boldsymbol{x} \in \mathcal{S}} C_{\boldsymbol{x}}^{\text{obj}}}{|\mathcal{S}|} \tag{5}$$

where $\mathcal{S}$ is the obtained non-dominated solution set. The convergence metric C^{obj} is calculated using

$$C_{\boldsymbol{x}}^{\text{obj}} = \sum_{i=1}^{m} \bar{f}_i(\boldsymbol{x}) \tag{6}$$

where $\bar{f}_i(\boldsymbol{x})$ represents the i-th normalized objective function value of solution $\boldsymbol{x}$. Then, all solutions in the solution set that satisfy the following

$$C^{\text{obj}} \leqslant \eta \times \text{obj}^{\text{average}} \tag{7}$$

are in the LR and $\eta > 1$ is a threshold (enlarge scale) to control the size of the LR.

The rationale of the proposed η threshold-based LR technique is illustrated in Fig. 5. In Fig. 5a, the red line (i.e., the PF) can represent $\text{obj}^{\text{average}}$ because all non-dominated solutions are on the PF, and the green dashed line can represent $\eta \times \text{obj}^{\text{average}}$. Thus, the shaded area is the LR. In the decision space, the function of LR can be illustrated in Fig. 5b. The η threshold further restricts the LR and reduces the search space to explore. The PR and LR provide a two-round selection. In the first round, solutions in PR are all selected, and in the second round, solutions in both PR and LR are selected.

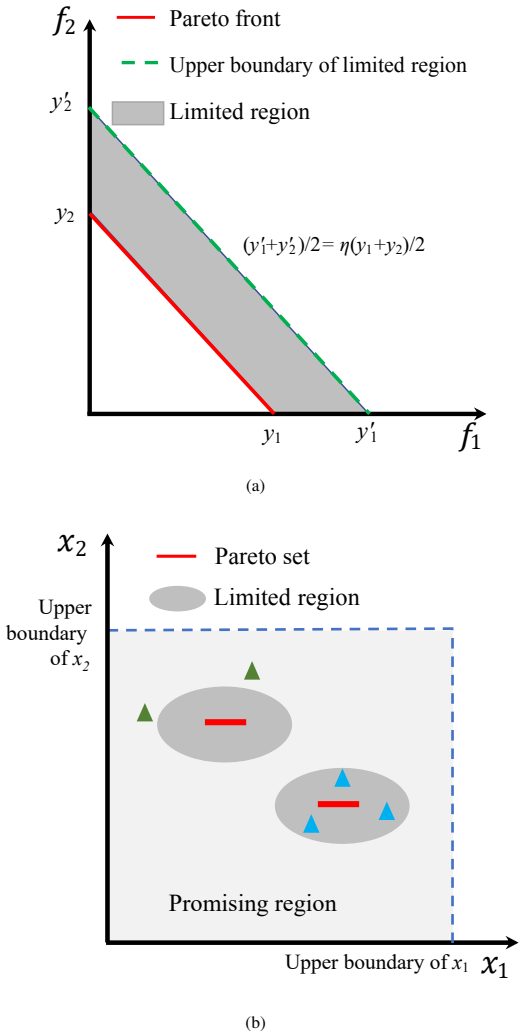


Fig. 5 Illustration of the proposed η threshold and the LR. Notably, the objective space has been normalized.

The reason why directly restricting LR is not suitable can also be illustrated by Fig. 5. As shown in Fig. 5b, if the PR is not first used, then the solutions outside the LR (green triangles) will be eliminated and only solutions in the LR (blue triangles) can be preserved. Consequently, the upper PS is less likely to be detected. On the contrary, if we select solutions in PR in the first round, then these solutions are preserved. Thereafter, these solutions (green triangles) can possibly be selected in the second round if the number of solutions in the LR (blue triangles) does not meet the required size. This situation is highly possible in the early stage of evolution as most solutions are not near the PSs.

Figure 6 exhibits the function of restricting LR under the circumstance of Fig. 4b. In this figure, we use the gray dashed lines to represent all the areas that satisfy

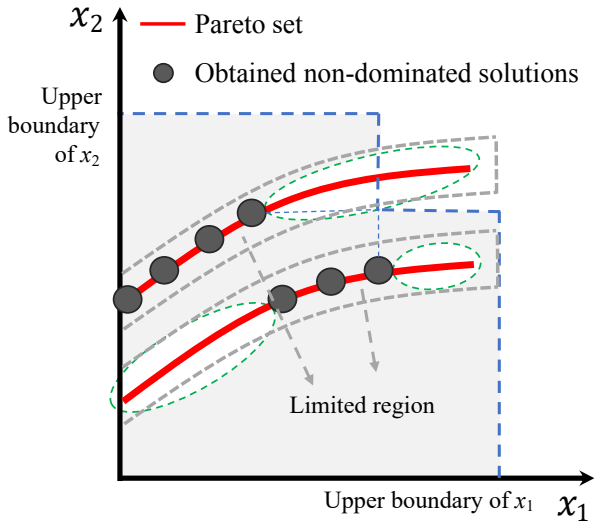


Fig. 6 Illustration of the function of the proposed limited region under the most general situation.

Formula (7), i.e., LR. The intersections of LR and PR contain many undetected areas on PSs.

3.2 Neighbor distance

The proposed neighbor distance can be calculated through the procedure of Algorithm 1. First, the Euclidean distances from a solution $\boldsymbol{x}$ to other solutions in the solution set $\mathcal{S}$ are calculated (line 3) and sorted in ascending order (line 4). Subsequently, D_x^{neighbor} is obtained by summing the former ns distance values (line 5). The Euclidean distance is formulated as follows:

$$\text{dist}(\boldsymbol{x},\boldsymbol{y}) = \sqrt{\sum_{i=1}^{n}(f_i(\boldsymbol{x}) - f_i(\boldsymbol{y}))^2} \tag{8}$$

where $i = 1, 2, \ldots, n$ denotes the i-th decision variable. Notably, f_i denotes the value of the i-th decision variable in Eq. (8).

To better illustrate the advantages of neighbor distance compared with crowding distance in Fig. 7, we create an artificial scene wherein the PSs are nonfunctional (two symmetrical segments). In the created scene, we need to select two solutions from E, F, G, and H. When using the crowding distance, the comparison is obtained as follows:

$$D^{\text{crowding}}(\boldsymbol{E}) = (x_1^E - x_1^f) + (x_1^F - x_1^E),$$
$$D^{\text{crowding}}(\boldsymbol{F}) = (x_1^F - x_1^e) + (x_1^g - x_1^F),$$
$$\because x_1^f = x_1^e \text{ and } x_1^E = x_1^F,$$
$$\therefore D^{\text{crowding}}(\boldsymbol{E}) = D^{\text{crowding}}(\boldsymbol{F}).$$

Similarly, $D^{\text{crowding}}(\boldsymbol{G}) = D^{\text{crowding}}(\boldsymbol{H})$.

However, if we use the proposed neighbor distance, the comparison is as follows:

$$D^{\text{neighbor}}(\boldsymbol{E}) = \text{dist}(\boldsymbol{E},\boldsymbol{e}) + \text{dist}(\boldsymbol{E},\boldsymbol{g}) + (x_1^E - x_1^f),$$
$$D^{\text{neighbor}}(\boldsymbol{F}) = \text{dist}(\boldsymbol{F},\boldsymbol{f}) + (x_1^F - x_1^e) + (x_1^g - x_1^F),$$

Algorithm 1 Calculation of the neighbor distance

Input: $\mathcal{S}$ (solution set), ns (neighbor size)
Output: D^{neighbor} (neighbor distance)

1: $D^{\text{neighbor}} \leftarrow \varnothing$;
2: **for** each solution $\boldsymbol{x} \in \mathcal{S}$ **do**
3: $d \leftarrow$ Calculate the Euclidean distance from x to other solutions in $\mathcal{S}$;
4: $d \leftarrow$ Sort d in ascending order;
5: $D_x^{\text{neighbor}} \leftarrow \sum_{i=1}^{\text{ns}} d_i$;
6: **end for**
7: **return** D^{neighbor}

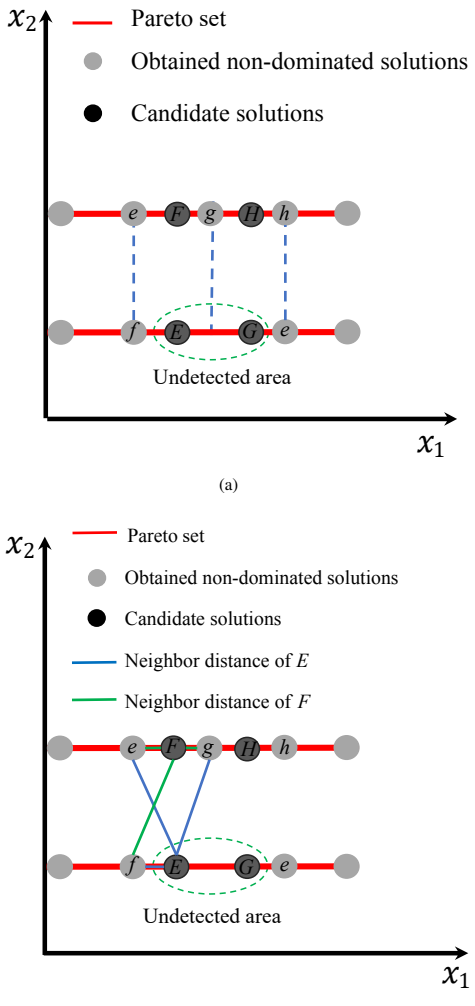


Fig. 7 Illustration of the advantages of the neighbor distance compared with the crowding distance. Solutions A and C have the same performance as B and D using crowding distance as the diversity metric (a), and have better performances using the proposed neighbor distance as the metric (b).

$\because (x_1^F - x_1^e) = (x_1^E - x_1^f)$ and $\text{dist}(\boldsymbol{F},\boldsymbol{f}) = \text{dist}(\boldsymbol{E},\boldsymbol{e})$,
$\therefore D^{\text{neighbor}}(\boldsymbol{E}) > D^{\text{neighbor}}(\boldsymbol{F})$.
Similarly, $D^{\text{neighbor}}(\boldsymbol{G}) > D^{\text{neighbor}}(\boldsymbol{H})$.

Thus, the neighbor distance can correctly evaluate the sparsity of solutions in dealing with the non-functional geometry of PSs.

3.3 MMOPR algorithm

3.3.1 Framework

Based on the above-mentioned techniques, we propose a novel dual-population-based coevolutionary MMOEA named MMOPR. The MMOPR procedure is illustrated in the flowchart in Fig. 8. In MMOPR, Population 1 (the convergence-prior-guided population) and Population 2 (the PR-guided population) coevolve. The mating selections of these two populations use different metrics, and the environmental selections use different priorities. Finally, Population 1 is output as the final solution set.

The pseudocode of the MMOPR framework is presented in Algorithm 2. The inputs include two parameters η and ns, and they are set to 2 and $\sqrt{N}$, respectively. After the initialization of the two populations, $\mathcal{P}_1$ and $\mathcal{P}_2$ (line 1), the neighbor distance and convergence performance must be calculated for mating selections (lines 2 and 3). In the main loop, the following steps are performed. First, the mating pools $\mathcal{M}_1$ and $\mathcal{M}_2$ of $\mathcal{P}_1$ and $\mathcal{P}_2$ are selected through Algorithm 3 based on D^{neighbor} and C^{obj}, respectively (lines 7 and 9). Additionally, their offspring sets are generated by the genetic algorithm (GA) operator (lines 8 and 10). Furthermore, $\mathcal{P}_1$ is updated by the convergence-prior-guided environmental selection using Algorithm 4, while $\mathcal{P}_2$ is updated by the PR-guided environmental selection using Algorithm 5.

3.3.2 Mating selection

The mating selections use double-tournament selection as the basis, and the pseudocode is presented in Algorithm 3. At each time two solutions are randomly selected (line 3), and the one with a better value of the comparison criterion is added into $\mathcal{M}$ (lines 4–7). If they have the same value according to the comparison criterion, we randomly add one to $\mathcal{M}$ (line 9). The tournament selection strategy can guarantee that all solutions, including better and worse-performing ones, are possibly selected. Furthermore, those elite solutions (with a considerably outstanding value of the comparison criterion) may be repeatedly selected. Thus, excellent

Algorithm 2 Framework of MMOPR

Input: N (population size), $G_{\max}$ (termination condition), η (enlarge scale of limited region), ns (neighbor size)
Output: $\mathcal{P}$ (final solution set)
1: $\mathcal{P}_1, \mathcal{P}_2 \leftarrow$ Generate the initial populations randomly;
2: $D^{\text{neighbor}} \leftarrow$ Calculate the neighbor distances of solutions in $\mathcal{P}_1$ by Algorithm 1;
3: $C^{\text{obj}} \leftarrow$ Calculate the convergence performances of solutions in $\mathcal{P}_2$ by Eq. (6);
4: $g \leftarrow$ Set the current generation zero;
5: **while** $t < G_{\max}$ **do**
6: $t \leftarrow t + 1$;
7: $\mathcal{M}_1 \leftarrow$ Select the mating pool of $\mathcal{P}_1$-based on D^{neighbor} by Algorithm 3;
8: $\mathcal{O}_1 \leftarrow$ Generate the offspring solution set of $\mathcal{P}_1$-based on the GA operator;
9: $\mathcal{M}_2 \leftarrow$-Select the mating pool of $\mathcal{P}_2$-based on C^{obj} by Algorithm 3;
10: $\mathcal{O}_2 \leftarrow$ Generate the offspring solution set of C $\mathcal{P}_2$-based on the GA operator;
11: $\mathcal{P}_1, D^{\text{neighbor}} \leftarrow$ Update $\mathcal{P}_1$ by the convergence-prior guided environmental selection by Algorithm 4 using $\mathcal{P}_1 \cup \mathcal{O}_1 \cup \mathcal{O}_2$;
12: $\mathcal{P}_2, C^{\text{obj}} \leftarrow$ Update $\mathcal{P}_2$ by the promising region guided environmental selection by Algorithm 5 using $\mathcal{P}_2 \cup \mathcal{O}_1 \cup \mathcal{O}_2$;
13: **end while**
14: **return** $\mathcal{P}_1$

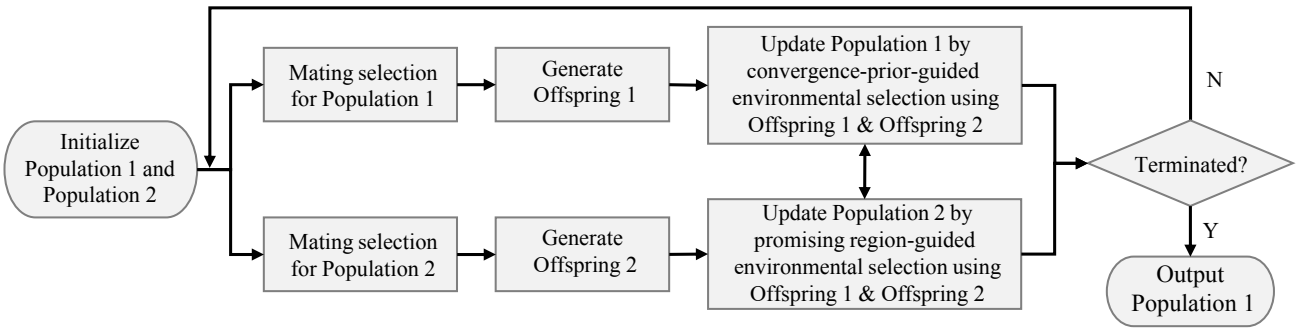


Fig. 8 Flowchart of the MMOPR procedure. The two populations generate offspring independently and evolve using Offspring 1 and Offspring 2.

Algorithm 3 Mating selection

Input: N (required size), $\mathcal{C}$ (candidate solution set), $\boldsymbol{K}$ (comparison criterion)
Output: $\mathcal{M}$ (selected mating pool)

$\mathcal{M} \leftarrow \varnothing$;
while $|\mathcal{M}| \leqslant N$ **do**
  $\boldsymbol{x}, \boldsymbol{y} \leftarrow$ Randomly select two solution from $\mathcal{C}$;
  if $\boldsymbol{K}_x > \boldsymbol{K}_y$ **then**
    $\mathcal{M} \leftarrow \mathcal{M} \cup \boldsymbol{y}$;
  else if $\boldsymbol{K}_x < \boldsymbol{K}_y$ **then**
    $\mathcal{M} \leftarrow \mathcal{M} \cup \boldsymbol{x}$;
  else
    $\mathcal{M} \leftarrow$ Randomly add $\boldsymbol{x}$ or $\boldsymbol{y}$ to $\mathcal{M}$;
  end if
end while
return $\mathcal{M}$

Algorithm 4 Convergence-prior-guided environmental selection

Input: N (required size), $\mathcal{C}$ (candidate solution set)
Output: $\mathcal{P}_1$ (updated solution set)

$F \leftarrow$ Determine the non-dominated front of solutions in $\mathcal{C}$ by fast non-dominated sorting;
$D^{\text{neighbor}} \leftarrow$ Calculate the neighbor distances of solutions in $\mathcal{C}$ by Algorithm 1;
$\mathcal{C} \leftarrow$ Sort solutions in $\mathcal{C}$ based on $\boldsymbol{K}$ in ascending order;
$\mathcal{C} \leftarrow$ Further sort solutions in $\mathcal{C}$ based on D^{neighbor} in ascending order;
$\mathcal{P}_1 \leftarrow$ Select the former N solutions;
return $\mathcal{P}_1$

Algorithm 5 Promising region guided environmental selection

Input: N (required size), $\mathcal{C}$ (candidate solution set), $\mathcal{P}_1$ (updated solution set), η (enlarge scale of limited region)
Output: $\mathcal{P}_2$ (updated solution set)

$\mathcal{P}_2 \leftarrow \varnothing$;
$\mathcal{P}_2 \leftarrow$ Preselect those solutions of $\mathcal{C}$ that locate in the promising region determined by Definition 3.1;
$\text{obj}_{\text{average}} \leftarrow$ Determine the average summary of objective values of solutions in $\mathcal{P}_1$ by Eq. (5);
$C^{\text{obj}} \leftarrow$ Calculate the convergence performances of solutions in $\mathcal{P}$ by Eq. (6);
$\mathcal{P}_2 \leftarrow$ Select solutions of $\mathcal{P}$ that satisfy Formula (7), i.e., in the Limited Region;
$\mathcal{C} \leftarrow \mathcal{C} \setminus \mathcal{P}_2$;
while $|\mathcal{P}_2| < N$ **do**
  $D^{\text{neighbor}} \leftarrow$ Calculate the neighbor distances of solutions in $\mathcal{C}$ by Algorithm 1;
  $\boldsymbol{x} = \arg\max_{\boldsymbol{x} \in \mathcal{C}} D_{\boldsymbol{x}}^{\text{neighbor}}$;
  $\mathcal{P}_2 \leftarrow \mathcal{P}_2 \cup \boldsymbol{x}$;
end while
while $|\mathcal{P}_2| > N$ **do**
  $D^{\text{neighbor}} \leftarrow$ Calculate the neighbor distances of solutions in $\mathcal{P}_2$ by Algorithm 1;
  $\boldsymbol{x} = \arg\min_{\boldsymbol{x} \in \mathcal{P}_2} D_{\boldsymbol{x}}^{\text{neighbor}}$;
  $\mathcal{P}_2 \leftarrow \mathcal{P}_2 \setminus \boldsymbol{x}$;
end while
return $\mathcal{P}_2$

genes can be better utilized. For the convergence-prior population $\mathcal{P}_1$, the neighbor distance is adopted as the comparison criterion. This is because such solutions distributed in sparse areas in the decision space can aid in exploring and exploiting PSs. As a consequence of the fact that convergence is preferred in environmental selection, preferring decision space diversity in the mating strategy can better balance convergence and decision space diversity. The PR-guided population $\mathcal{P}_2$ adopts the convergence performance C^{obj} as the criterion because convergence is ignored in the PR-guided environmental selection. Thus, preferring solutions with better objective function values can accelerate the approximation to the LR and reduce the function evaluations in invalid areas that are far from PSs.

According to the framework, mating selections and the generation of offspring sets for $\mathcal{P}_1$ and $\mathcal{P}_2$ are conducted separately. In a small decision space, independent mating and mixed mating may have very small differences. However, given that the mating strategy of $\mathcal{P}_1$ prefers solutions on PSs while the mating strategy of $\mathcal{P}_2$ considers solutions in the LR (mostly not on the PSs), mixed mating may generate many useless offspring between the PS and outside areas if the decision space is larger.

3.3.3 Convergence-prior-guided environmental selection

The pseudocode of convergence-prior-guided environmental selection is presented in Algorithm 4. Specifically, two rankings of all solutions in the candidate solution set (i.e., $\mathcal{P}_1 \cup \mathcal{O}_1 \cup \mathcal{O}_2$) are obtained first (lines 1 and 2): the non-dominated front and the neighbor distance. The non-dominated front is prioritized (line 3) so $\mathcal{P}_1$ prefers non-dominated solutions. Then, the solution set is secondarily sorted by neighbor distance (line 4); consequently, solutions in the same non-dominated front will be sorted. Finally, the former N solutions are selected for $\mathcal{P}_1$. Notably, objective space diversity is not considered in this selection strategy. This is because $\mathcal{P}_1$, which prefers convergence, will soon be filled with non-dominated solutions on PF and PSs. On this basis, the PSs will be well-covered if the decision space diversity is well-

enhanced. Hence, the PF can also be well-covered. In the experiments, we evaluated this through comparative studies in terms of objective space convergence and diversity.

3.3.4 Promising region guided environmental selection

The pseudocode of PR-guided environmental selection is presented in Algorithm 5. The candidate solution set of this procedure is $\mathcal{P}_2 \cup \mathcal{O}_1 \cup \mathcal{O}_2$. Two rounds of preselection are performed in this selection procedure. In the first round, solutions in the PR are preselected (line 2), after which solutions in the LR are preselected (lines 3–5) in the second round. If the number of preselected solutions in $\mathcal{P}_2$ is not enough (less than N), then the solutions with the best neighbor distance are selected one by one from the remainder until the number meets N (lines 7–11). In the earlier evolutionary process, many solutions may be distributed outside the LR. By this deletion strategy, the diversity in the PR can be guaranteed, and thus, more LRs can be detected then. On the contrary, if the size exceeds N, then, solutions with the worst neighbor distance are deleted one by one until the size meets N (lines 12–16). The two rounds of preselection guarantee that the preselected solutions are in LR. Thus, the deletion strategy can enhance the decision space diversity by the neighbor distance.

3.4 Computational complexity

Suppose N is the population size, and m and n denote the number of objectives and decision variables, respectively. The computational complexity of calculating neighbor distance is $O(nN^2)$, while the computational complexity of using Definition 3.1 is $O(nN)$. The convergence-prior-guided environmental selection consumes $O(mN^2)$ for the fast non-dominated sorting, and $O(nN^2)$ for calculating the neighbor distance. The PR-guided environmental selection consumes $O(n(N)^2)$ for calculating the neighbor distance in the deletion strategies. Therefore, the overall computational complexity is $O(nN^2)$ or $O(mN^2)$, which is determined by the values of n and m (i.e., the decision variable and the objective function dimensions).

4 Experimental Study

This section presents the details of our experimental studies, including the experimental settings in Section 4.1, the comparison studies in Section 4.2, the ablation and parameter analysis in Section 4.3, and the discussions on the proposed neighbor distance in Section 4.4. All experiments are conducted on PlatEMO[30].

4.1 Experimental settings

4.1.1 Benchmark problems

Three widely used MMOP benchmarks are used in the experiments: MMF[9], IDMP[25], and MMMOP[13]. The parameters, features, and challenges of these benchmark MMOPs are presented in detail in Section S2.3 to Section S2.5 in the ESM.

4.1.2 Algorithms in comparison

Six state-of-the-art MMOEAs are chosen for comparison studies. These include three state-of-the-art convergence-prior MMOEAs: DN-NSGA-II[11], MO-Ring-PSO-SCD[9], and TriMOEA-TA&R[13]; two highly advanced convergence-balanced MMOEAs: MMOEADC[12] and MMEAWI[14]; and one very recently proposed MMOEA: HREA[10].

4.1.3 Parameter settings

All the algorithms in this work use the operators of GA to generate offspring. The simulated binary crossover (SBC)[31] and polynomial mutation (PM)[15] are used with the following parameter settings:

- The crossover probability is $p_c = 1$; the distribution index is $\eta_c = 20$;
- The mutation probability is $p_m = 1/n$; the distribution index is $\eta_m = 20$.

The parameter settings about population size N, maximum iterations G_{max}, and maximum evaluations E_{max} are presented in Table S1 in the ESM. All algorithms in comparison and benchmark problems adopt the same parameters as their original settings, i.e., the default settings in PlatEMO. The MMOPR parameters include the η and ns used in the proposed LR determination and neighbor distance. In MMOPR, they are set to 2 and $\sqrt{N}$. Their parameter analysis will be studied in detail in Section 4.3.

4.1.4 Performance indicator

In this work, the IGDX indicator[11] is adopted as the performance indicator to evaluate the performance of MMOPR and other methods in terms of decision space convergence and diversity. The IGDX indicator can be regarded as IGD[32] in the decision space. To calculate IGDX, a set of uniformly distributed reference points are generated on the true PS of an MMOP instance according to the approach of Ref. [33]. Then, the distances of each reference point to its nearest solution among the solution set are summed as the IGDX value. A smaller IGDX

value indicates a better result. In addition, the HV[32] indicator is applied to evaluate the performance in the objective space due to its Pareto-compliant advantage. Here, $(1.1, \ldots, 1.1)$ is used as the reference point in the normalized objective space to calculate the HV value. The CR[34] indicator is also adopted to measure the cover rate of the obtained solution set on the true PSs in the decision space to evaluate the spread. The three indicators are all used in this work to achieve a fair comparison[35]. Only IGDX and CR are adopted for the MMMOP test suite in this work. This is because this test suite lacks some necessary information, which results in NaN values in calculating the HV indicator. In general, the HV indicator can measure the convergence and objective space diversity, while the IGDX and CR indicators can measure the decision space diversity.

4.1.5 Statistical analyses

Each algorithm executes 30 independent runs on each test instance. The mean and standard deviation values of HV, IGDX, and CR are recorded. The Friedman test with the Holm correction at a significance level of 0.05 and the Wilcoxon rank-sum test with a significance level of 0.05 are used to perform the statistical analysis by means of the KEEL software[36]. Additionally, "+" "−", and "≈" are used to show that the result of other algorithms is significantly better than, significantly worse than, and statistically similar to those obtained by MMOPR according to the Wilcoxon test, respectively.

4.2 Comparison studies

In the first part of the experiments, we compare the MMOPR with the selected six state-of-the-art MMOEAs on the selected benchmarks. Guidance on how to read the results on PF and PS in this paper is provided in Section S2.2 in the ESM.

The HV, IGDX, and CR results on the MMF test suite are reported in Tables 1–3. On the MMF test suite, MMOPR obtains the best overall results. The final PSs obtained by MMOPR are depicted in Fig. 1, which shows that the MMOPR can identify all the PSs and achieve a good degree of diversity. In Fig, 4, we depict the PSs obtained by all algorithms on MMF3, MMF7, and MMF8. Through the detailed comparisons, it can be clearly seen that the final solution set of MMOPR is evenly distributed on the PSs, revealing that it obtains the best convergence, objective space diversity, and decision space diversity.

The results of HV, IGDX, and CR on the IDMP test suite are reported in Tables S5–S7 in the ESM. Given

Table 1 Statistical results of HV obtained by MMOPR and other methods on MMF benchmark problems. The best result in each row is highlighted. "+" "−", and "≈" are used to show that the result of other algorithms was significantly better than, significantly worse than, and statistically similar to those obtained by MMOPR according to the Wilcoxon test, respectively.

Problem	DNNSGAII	MO-Ring-PSO-SCD	TriMOEATAR	MMEAWI	MMOEADC	HREA	MMOPR
MMF1	0.9068 (0.0001) ≈	0.9058 (0.0006) −	0.9058 (0.0020) −	0.9065 (0.0003) −	0.9053 (0.0011) −	0.9053 (0.0015) −	0.9068 (0.0001)
MMF2	0.8585 (0.0028) −	0.8481 (0.0032) −	0.8557 (0.0035) −	0.8596 (0.0002) −	0.8594 (0.0004) −	0.8585 (0.0010) −	0.8600 (0.0001)
MMF3	0.8122 (0.0021) −	0.8019 (0.0041) −	0.8081 (0.0034) −	0.8131 (0.0003) −	0.8126 (0.0004) −	0.8122 (0.0005) −	0.8136 (0.0001)
MMF4	0.7219 (0.0012) ≈	0.7223 (0.0002) ≈	0.7017 (0.0387) −	0.7221 (0.0001) ≈	0.7231 (0.0002) +	0.7212 (0.0010) −	0.7220 (0.0007)
MMF5	0.9687 (0.0002) ≈	0.9687 (0.0001) ≈	0.9686 (0.0008) −	0.9688 (0.0001) +	0.9683 (0.0003) −	0.9685 (0.0003) −	0.9686 (0.0004)
MMF6	0.9531 (0.0003) ≈	0.9531 (0.0002) ≈	0.9530 (0.0011) −	0.9532 (0.0001) ≈	0.9525 (0.0006) −	0.9525 (0.0006) −	0.9531 (0.0005)
MMF7	0.8841 (0.0001) ≈	0.8837 (0.0002) −	0.8838 (0.0011) −	0.8836 (0.0002) −	0.8832 (0.0015) ≈	0.8832 (0.0009) −	0.8841 (0.0001)
MMF8	0.9703 (0.0004) ≈	0.9705 (0.0002) ≈	0.9693 (0.0083) −	0.9702 (0.0004) ≈	0.9703 (0.0005) ≈	0.9699 (0.0007) −	0.9704 (0.0003)
+/ − / ≈	0/2/6	0/4/4	0/8/0	1/4/3	1/5/2	0/8/0	–

Table 2 Statistical results of IGDX obtained by MMOPR and other methods on MMF benchmark problems. The best result in each row is highlighted. "+" "−", and "≈" are used to show that the result of other algorithms was significantly better than, significantly worse than, and statistically similar to those obtained by MMOPR according to the Wilcoxon test, respectively.

Problem	DNNSGAII	MO-Ring-PSO-SCD	TriMOEATAR	MMEAWI	MMOEADC	HREA	MMOPR
MMF1	0.0406 (0.0009) −	0.0707 (0.0129) −	0.0487 (0.0028) −	0.0391 (0.0021) ≈	0.0440 (0.0020) −	0.0442 (0.0031) −	0.0386 (0.0010)
MMF2	0.0405 (0.0297) −	0.0529 (0.0186) −	0.0676 (0.0497) −	0.0127 (0.0041) −	0.0084 (0.0014) ≈	0.0129 (0.0046) −	0.0082 (0.0007)
MMF3	0.0238 (0.0115) −	0.0422 (0.0224) −	0.0392 (0.0146) −	0.0098 (0.0029) −	0.0081 (0.0017) ≈	0.0095 (0.0014) −	0.0079 (0.0006)
MMF4	0.0266 (0.0022) −	0.0598 (0.0135) −	0.1197 (0.1679) −	0.0235 (0.0010) ≈	0.0218 (0.0017) +	0.0289 (0.0021) −	0.0236 (0.0013)
MMF5	0.0815 (0.0045) −	0.1268 (0.0159) −	0.0880 (0.0079) −	0.0688 (0.0024) +	0.0778 (0.0033) −	0.0761 (0.0053) ≈	0.0751 (0.0046)
MMF6	0.0712 (0.0041) −	0.1039 (0.0142) −	0.0764 (0.0054) −	0.0620 (0.0017) +	0.0677 (0.0025) −	0.0690 (0.0046) −	0.0648 (0.0022)
MMF7	0.0219 (0.0008) ≈	0.0508 (0.0116) −	0.0370 (0.0266) −	0.0235 (0.0014) −	0.0245 (0.0040) ≈	0.0243 (0.0019) −	0.0220 (0.0011)
MMF8	0.0670 (0.0127) −	0.1665 (0.0695) −	0.4435 (0.0993) −	0.0519 (0.0056) −	0.0482 (0.0080) ≈	0.0611 (0.0047) −	0.0480 (0.0046)
+/ − / ≈	0/7/1	0/8/0	0/8/0	2/4/2	1/3/4	0/7/1	–

Table 3 Statistical results of CR obtained by MMOPR and other methods on MMF benchmark problems. The best result in each row is highlighted. "+" "−", and "≈" are used to show that the result of other algorithms was significantly better than, significantly worse than, and statistically similar to those obtained by MMOPR according to the Wilcoxon test, respectively.

Problem	DNNSGAII	MO-Ring-PSO-SCD	TriMOEATAR	MMEAWI	MMOEADC	HREA	MMOPR
MMF1	0.9992 (0.0008) −	0.9824 (0.0150) −	0.9923 (0.0052) −	0.9987 (0.0016) −	0.9952 (0.0029) −	0.9966 (0.0030) −	0.9998 (0.0004)
MMF2	0.9688 (0.0486) −	0.9163 (0.0443) −	0.9375 (0.0607) −	0.9988 (0.0029) +	0.9972 (0.0080) −	0.9880 (0.0212) −	0.9978 (0.0077)
MMF3	0.9553 (0.0448) −	0.9198 (0.0481) −	0.8900 (0.0683) −	0.9970 (0.0095) −	0.9991 (0.0029) +	0.9902 (0.0117) −	0.9985 (0.0059)
MMF4	0.9999 (0.0003) ≈	0.9784 (0.0190) −	0.9317 (0.1260) −	0.9992 (0.0013) −	0.9988 (0.0008) −	0.9969 (0.0044) −	1.0000 (0.0000)
MMF5	0.9990 (0.0018) −	0.9783 (0.0189) −	0.9928 (0.0084) −	0.9989 (0.0014) −	0.9958 (0.0026) −	0.9972 (0.0036) −	0.9996 (0.0009)
MMF6	0.9987 (0.0021) −	0.9785 (0.0180) −	0.9945 (0.0073) −	0.9991 (0.0009) −	0.9960 (0.0031) −	0.9974 (0.0023) −	0.9997 (0.0005)
MMF7	0.9994 (0.0010) ≈	0.9703 (0.0282) −	0.8865 (0.1829) −	0.9970 (0.0047) −	0.9965 (0.0017) −	0.9979 (0.0034) −	0.9996 (0.0007)
MMF8	0.9864 (0.0104) −	0.9594 (0.0267) −	0.8810 (0.0533) −	0.9969 (0.0020) ≈	0.9912 (0.0063) −	0.9950 (0.0049) ≈	0.9942 (0.0048)
+/ − / ≈	0/6/2	0/8/0	0/8/0	1/6/1	1/7/0	0/7/1	–

that the IDMP test suite has imbalanced distances in the objective and decision spaces, some algorithms (DNNSGA-II and MMEAWI) performs well on HV, and some algorithms (MMOEADC and HREA) performs well on IGDX. However, they are unable to achieve good results on both indicators (i.e., both spaces). On the contrary, MMOPR can achieve the best tradeoff between these two indicators even though it does not have the best performance on HV and IGDX. This finding reveals that MMOPR can handle the imbalanced distances in IDMP. The PSs obtained by MMOPR on IDMP instances are depicted in Fig. S2 in the ESM. Further, we select IDMPM2T3, IDMPM3T4, and IDMPM4T1, and the PSs obtained by all algorithms are depicted in Fig. 9. The results show that MMOPR and MMEAWI achieve the best convergence and diversity in these instances. Furthermore, HREA and MMOEADC obtain better IGDX values because they obtain better decision space diversity. However, their HV values are worse because they include some dominated solutions.

The results of IGDX and CR on the MMMOP test suite are reported in Tables S8 and S9 in the ESM. In general, MMEAWA obtains the best overall performance in terms of IGDX, and HREA obtains the best in CR. However, they perform poorly on the other indicator. In addition, MMOPR obtains the best overall performance of IGDX and CR. Figure S3 in the ESM depicts the PSs obtained by MMOPR on all instances of MMMOP. Clearly, MMOPR can find all the PSs even if there are eight PSs on MMMOP6C. Figure S5 in the ESM presents the PSs obtained by all algorithms on MMMOP1B, MMMOP4C, MMMOP5C, and MMMOP6C. Figure S5 in the ESM clearly shows that only MMOPR can find all the PSs on any instance. In summary, MMOPR obtains the best versatility on MMMOPs.

The convergence profiles of the IGDX values of all algorithms on MMF3, MMF7, MMF8, MMMOP1B, MMMOP4C, and MMMOP5C are presented in Fig. 10. As shown in Fig. 10, MMOPR obtains the best convergence speed and final values on MMF3, MMF7, MMF8, and MMMOP4C. In MMMOP1B and MMMOP5C, MMOPR also obtain the best final values.

Table 4 presents the average rankings of MMOPR and other methods on the CPU run time in all instances via the Friedman test. From the rankings, we can see that MMOPR is more time efficient than all the recently proposed methods (MMEAWI, MMOEADC, and HREA). This is because no clustering or decomposition is needed in MMOPR.

4.3 Ablation and parameter studies

In this part, we conduct ablation studies to investigate the effectiveness of different components and parameter studies to test the influence of different settings in our methods. The detailed features and functions of all variants are presented in Table 4. The results are reported in Tables S10–S17 in the ESM. The results reveal that MMOPR-P_1 generally performs worse than MMOPR in most instances, revealing that the PR-guided population is helpful. MMOPR-All also performs worse than MMOPR, demonstrating the necessity of LR. The results of MMOPR-η demonstrate that η must be larger than 1, while MMOPR-3η reveals that $\eta = 2$ has the better setting. MMOPR-MSD also demonstrates that using C^{obj} as the measure of the mating selection is more effective than other variants. Furthermore, MMOPR-$N/2$ and MMOPR-$N/10$ indicate that the influence of different neighbor sizes is mainly on IDMP (i.e., problems with imbalanced distances in the objective and decision spaces).

Figure 11 depicts the distributions of two populations

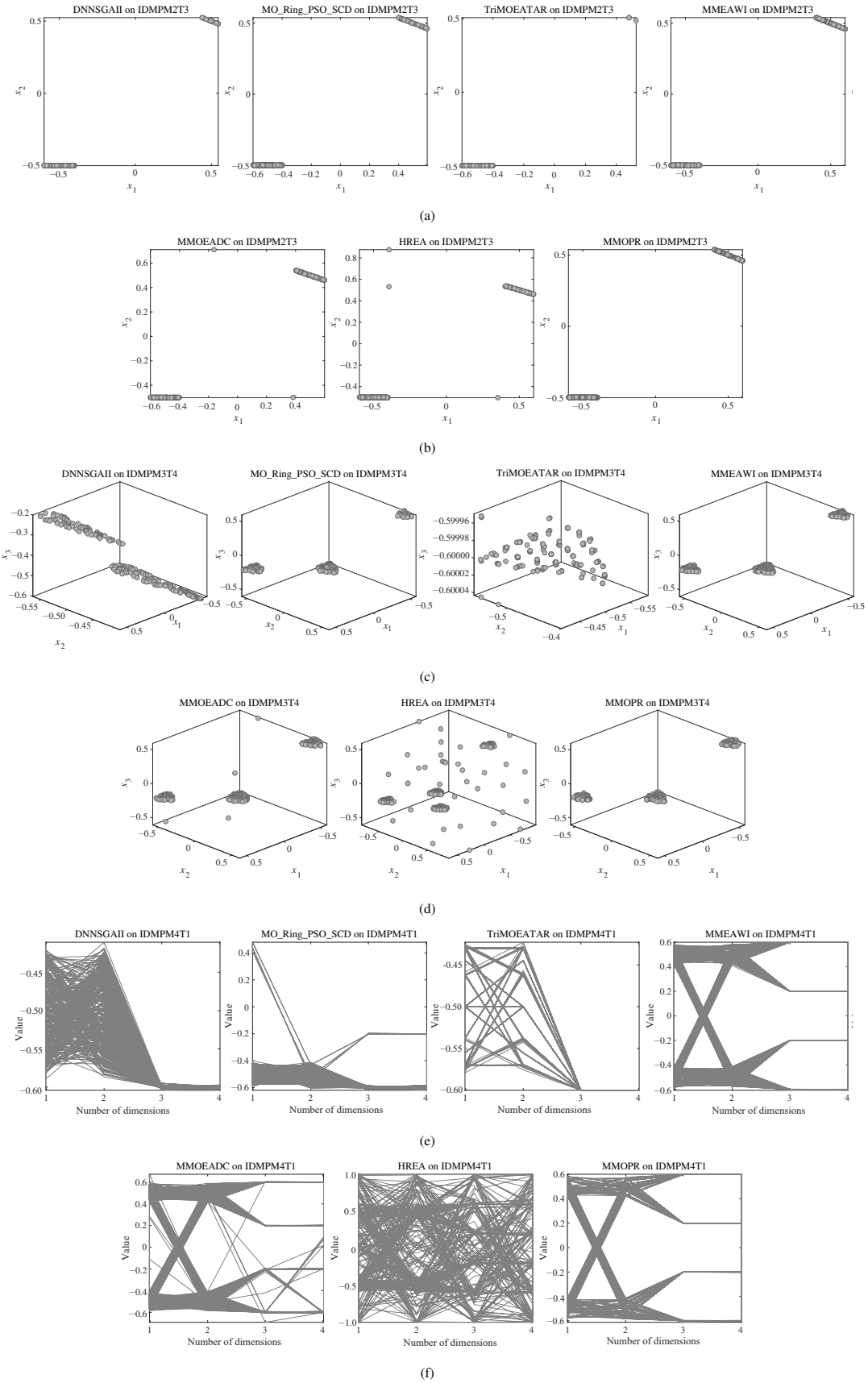
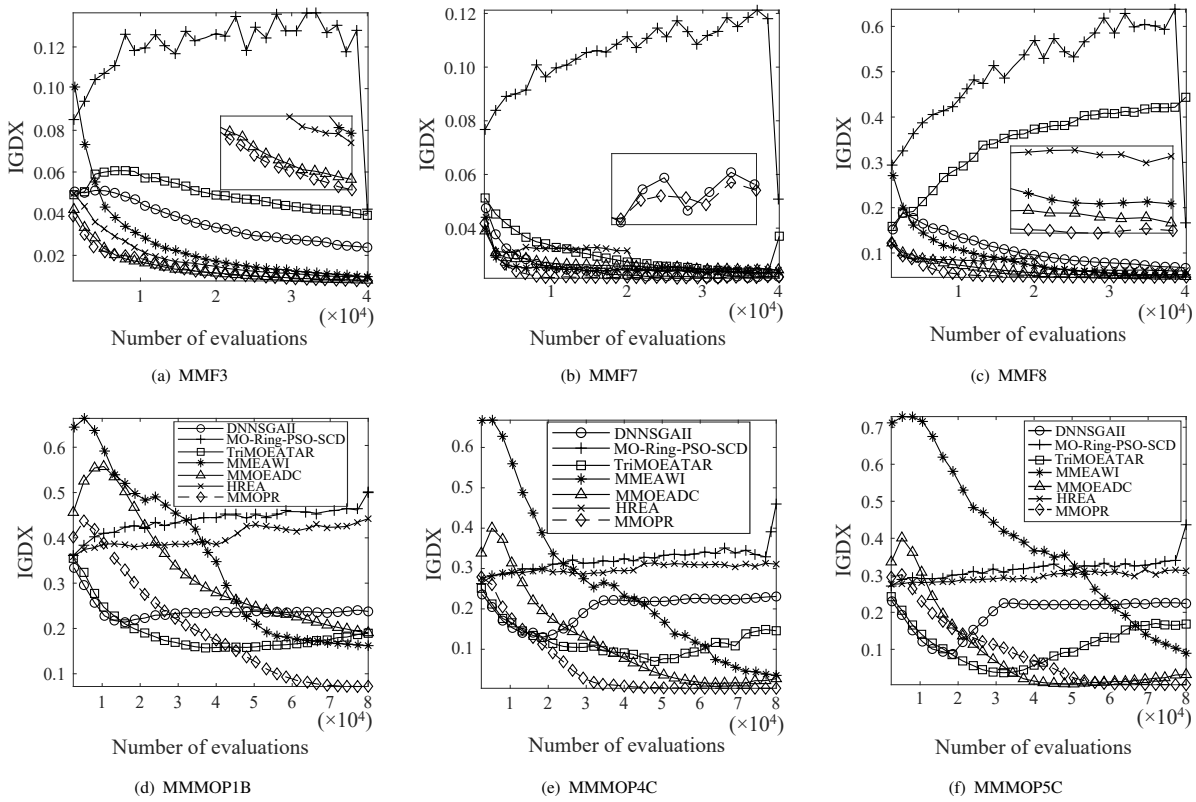


Fig. 9 Pareto sets obtained by MMOPR and other methods in dealing with IDMPM2T3, IDMPM3T4, and IDMPM4T1 with the median IGDX values of 30 runs.



(a) MMF3 (b) MMF7 (c) MMF8

(d) MMMOP1B (e) MMMOP4C (f) MMMOP5C

Fig. 10 Convergence profile of IGDX indicator values obtained by MMOPR and other methods in dealing with MMF3, MMF7, MMF8, MMMOP1B, MMMOP4C, and MMMOP5C with the median IGDX values of 30 runs.

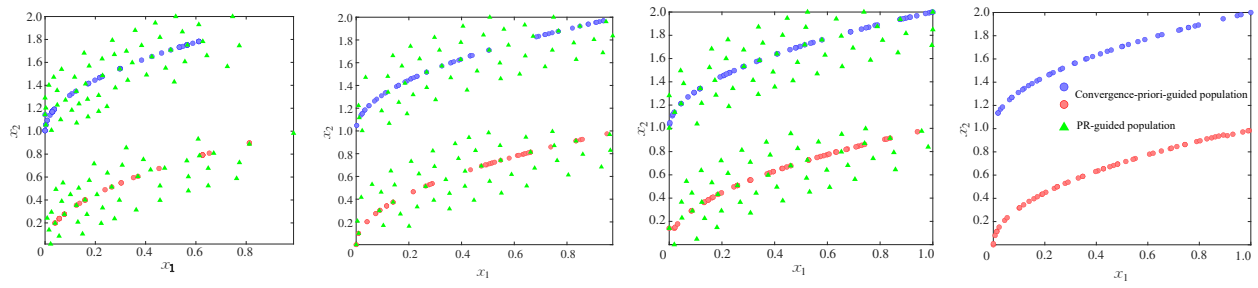
Table 4 Average rankings of MMOPR and other methods on the run time obtained using the Friedman test.

Algorithm	Ranking
DN-NSGA-II	1.250
MO-Ring-PSO-SCD	4.95
TriMOEAT&R	**2.125**
MMEAWI	6.125
MMOEADC	5.275
HREA	5.650
MMOPR	2.625

in MMOPR at different stages of the evolution, where the dots are solutions of the convergence-prior-guided population and the triangles are solutions of the PR-guided population. At early stage, the convergence-prior-guided population cannot detect the whole PSs, however, the PR-guided population can preserve useful solutions near the undetected areas. Therefore, the convergence-prior-guided population achieves better diversity on the PSs then.

4.4 Discussions on neighbor distance

In this part, we discuss the property of our proposed neighbor distance through comparison studies with the three other crowding distances in DN-NSGA-II, MO-Ring-PSO-SCD, and HREA, respectively. The variants are introduced in Table 4. The results are reported



(a) Populations at 50-th generation (b) Populations at 100-th generation (c) Populations at 150-th generation (d) Populations at the end of evolution

Fig. 11 Distributions of two populations at different stages of the evolutionary process. The blue and red dots represent solutions on different Pareto set, while the triangles are solutions of the PR-guided population.

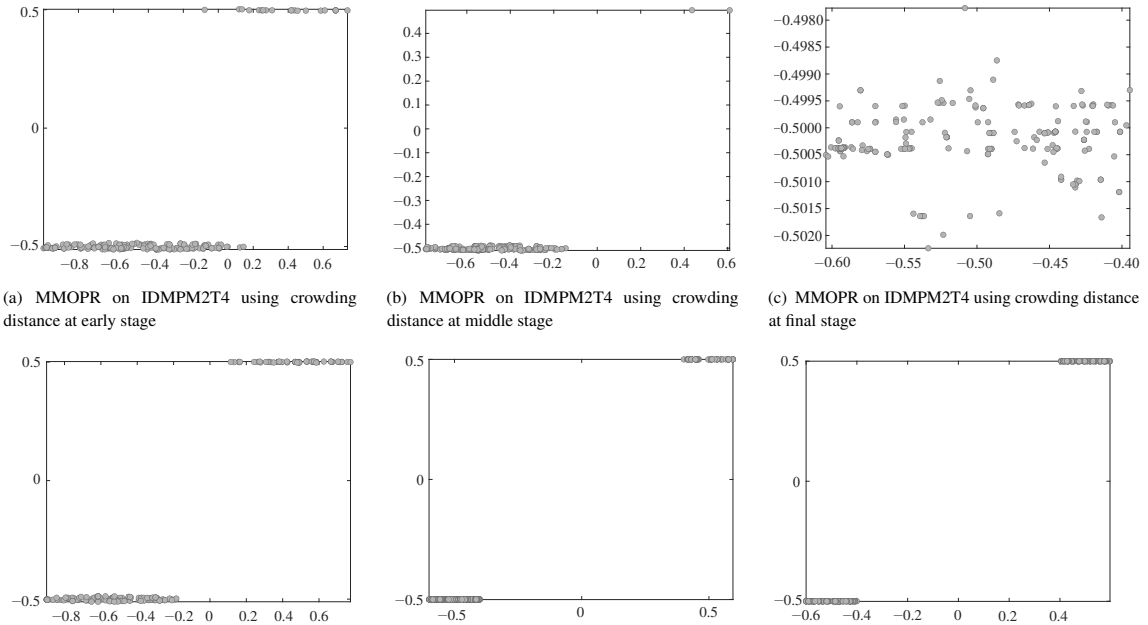
in Tables S18–S25 in the supplementary file. On the MMF test suite, MMOPR outperforms the three other variants on all indicators, revealing that the neighbor distance is useful in MMFs. On the IDMP test suite, only MMOPR-DN performs better on HV but performs significantly worse on IGDX. On the MMMOP test suite, MMOPR obtains at least competitive results. Therefore, the proposed neighbor distance is more effective than the three different crowding distances. In Fig. 12, we present the convergence process of the convergence-prior population in dealing with IDMPM2T4. Here, Figs. 12a–12c present the process of using the crowding distance of DN-NSGA-II, while Figs. 12d–12f present the process of using the neighbor distance. When the crowding distance is used, the solutions on the PS in the upper right corner cannot be detected as sparse region solutions and are thus deleted. However, if we use the proposed neighbor distance, they can be preserved and the upper right corner PS can be found.

5 Conclusion and Future Work

In this article, we explore a so-called LR in a PR, where PSs can be fully or partially included in the decision space to assist the search for PSs. We also propose a neighbor distance measure to enhance the decision space diversity, thereby overcoming the ineffectiveness of crowding distance in the complex geometry of PSs in the decision space. Based on these techniques, we propose a novel dual-population-based coevolutionary MMOEA named MMOPR.

In the experiments, we extensively evaluate the performance of MMOPR and the proposed neighbor distance. The results reveal that MMOPR achieves the best overall performance and versatility in dealing with different MMOPs. Furthermore, the neighbor distance can overcome the limitation of crowding distance in dealing with the complex geometry of PSs in the decision space. The results also reveal that our methods are more time efficient than most state-of-the-art MMOEAs.

Nevertheless, the results reveal that MMOPR lacks improvement in objective space diversity, resulting in poor HV values in IDMPs. In the future, enhancing the objective space diversity through some advanced techniques[37–39] is expected. Moreover, to better reflect the feature and challenges of real-world optimization problems, it is necessary to develop constrained MMOP benchmark test suites since most real-world optimization problems contain constraints[40–42]. Last but not least, given the fact that existing MMOP benchmarks are mostly not scaled, developing a scalable test suite[43] with multi-objective features[44, 45] or large-



(a) MMOPR on IDMPM2T4 using crowding distance at early stage

(b) MMOPR on IDMPM2T4 using crowding distance at middle stage

(c) MMOPR on IDMPM2T4 using crowding distance at final stage

(d) MMOPR on IDMPM2T4 using neighbor distance at early stage

(e) MMOPR on IDMPM2T4 using neighbor distance at middle stage

(f) MMOPR on IDMPM2T4 using neighbor distance at final stage

Fig. 12 Convergence process in the decision space, where the axes represent the first and the second decision vector dimension of the convergence-prior population in dealing with IDMPM2T4. Figures (a)–(c) use the crowding distance of DN-NSGA-II, Figs. (d)–(f) use the proposed neighbor distance.

scale decision vectors[3] is also expected.

The codes of MMOPR can be obtained from the authors upon request.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (No. 62076225).

Electronic Supplementary Material

Supplementary materials including

• Table S1: Parameters settings for MMOPR and other methods. N denotes the population size, $G_{\max}$ is the maximum number of iterations, and $E_{\max}$ is the maximum number of evaluations.

• Table S2: Parameters and main features of MMF test suite.

• Table S3: Parameters and features of MMMOP test suite.

• Table S4: Details for variants of MMOPR, including the name, feature, and function of each variant.

• Table S5: Statistical results of HV obtained by MMOPR and other methods on IDMP benchmark problems.

• Table S6: Statistical results of IGDX obtained by MMOPR and other methods on IDMP benchmark problems.

• Table S7: Statistical results of CR obtained by MMOPR and other methods on IDMP benchmark problems.

• Table S8: Statistical results of IGDX obtained by MMOPR and other methods on MMMOP benchmark problems.

• Table S9: Statistical results of CR obtained by MMOPR and other methods on MMMOP benchmark problems.

• Table S10: Statistical results of HV obtained by MMOPR and other methods on MMF benchmark problems.

• Table S11: Statistical results of IGDX obtained by MMOPR and other methods on MMF benchmark problems.

• Table S12: Statistical results of CR obtained by MMOPR and other methods on MMF benchmark problems.

• Table S13: Statistical results of HV obtained by MMOPR and other methods on IDMP benchmark problems.

• Table S14: Statistical results of IGDX obtained by MMOPR and other methods on IDMP benchmark problems.

• Table S15: Statistical results of CR obtained by MMOPR and other methods on IDMP benchmark problems.

• Table S16: Statistical results of IGDX obtained by MMOPR and other methods on MMMOP benchmark problems.

• Table S17: Statistical results of CR obtained by MMOPR and other methods on MMMOP benchmark problems.

• Table S18: Statistical results of HV obtained by MMOP and its variants using crowding distance metrics on MMMF benchmark problems.

• Table S19: Statistical results of IGDX obtained by MMOPR and its variants using crowding distance metrics on MMMF benchmark problems.

• Table S20: Statistical results of CR obtained by MMOPR and its variants using crowding distance metrics on MMMF benchmark problems.

• Table S21: Statistical results of HV obtained by MMOPR and its variants using crowding distance metrics on IDMP benchmark problems.

• Table S22: Statistical results of IGDX obtained by MMOPR and its variants using crowding distance metrics on IDMP benchmark problems.

• Table S23: Statistical results of CR obtained by MMOPR and its variants using crowding distance metrics on IDMP benchmark problems.

• Table S24: Statistical results of IGDX obtained by MMOPR and its variants using crowding distance metrics on MMMOP benchmark problems.

• Table S25: Statistical results of CR obtained by MMOPR and its variants using crowding distance metrics on MMMOP benchmark problems.

• Fig. S1: Final Pareto sets obtained by MMOPR on MMF benchmark problems with the median IGDX value of 30 runs.

• Fig. S2: Final Pareto sets obtained by MMOPR on IDMP benchmark problems with the median IGDX value of 30 runs.

• Fig. S3: Final Pareto sets obtained by MMOPR on MMMOP benchmark problems with the median IGDX value of 30 runs.

• Fig. S4: Pareto sets obtained by MMOPR and other methods in dealing with MMF3, MMF7, and MMF8 with the median IGDX values of 30 runs.

• Fig. S5: Pareto sets obtained by MMOPR and other methods in dealing with MMMOP1B, MMMOP4C, MMMOP5C, and MMMOP6C with the median IGDX values of 30 runs.

All the tables and figures are available in the online version of this article at https://doi.org/10.26599/TST.2023.9010031.par

References

[1] S. Han, K. Zhu, M. Zhou, and X. Cai, Competition-driven multimodal multiobjective optimization and its application to feature selection for credit card fraud detection, *IEEE Trans. Syst., Man, Cybern.: Syst.*, vol. 52, no. 12, pp. 7845–7857, 2022.

[2] V. Barichard and J. K. Hao, Genetic tabu search for the Multi-Objective knapsack problem, *Tsinghua Science and Technology*, vol. 8, no. 1, pp. 8–13, 2003.

[3] Y. Tian, R. Liu, X. Zhang, H. Ma, K. C. Tan, and Y. Jin, A multipopulation evolutionary algorithm for solving large-scale multimodal multiobjective optimization problems, *IEEE Trans. Evol. Comput.*, vol. 25, no. 3, pp. 405–418, 2021.

[4] H. Zhang, L. Ma, J. Wang, and L. Wang, Furnace-grouping problem modeling and multi-objective optimization for special aluminum, *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 6, no. 3, pp. 544–555, 2022.

[5] X. Ma, Y. Fu, K. Gao, L. Zhu, and A. Sadollah, A multi-objective scheduling and routing problem for home health care services via brain storm optimization, *Complex System Modeling and Simulation*, vol. 3, no. 1, pp. 32–46, 2023.

[6] L. Wang, Z. Pan, and J. Wang, A review of reinforcement learning based intelligent optimization for manufacturing scheduling, *Complex System Modeling and Simulation*, vol. 1, no. 4, pp. 257–270, 2021.

[7] R. Tanabe and H. Ishibuchi, A review of evolutionary multimodal multiobjective optimization, *IEEE Trans. Evol. Comput.*, vol. 24, no. 1, pp. 193–200, 2020.

[8] W. Gong, Z. Liao, X. Mi, L. Wang, and Y. Guo, Nonlinear equations solving with intelligent optimization algorithms: A survey, *Complex System Modeling and Simulation*, vol. 1, no. 1, pp. 15–32, 2021.

[9] C. Yue, B. Qu, and J. Liang, A multiobjective particle swarm optimizer using ring topology for solving multimodal multiobjective problems, *IEEE Trans. Evol. Comput.*, vol. 22, no. 5, pp. 805–817, 2018.

[10] W. Li, X. Yao, T. Zhang, R. Wang, and L. Wang, Hierarchy ranking method for multimodal multiobjective optimization with local Pareto fronts, *IEEE Trans. Evol. Comput.*, pp. vol. 27, no. 1, pp. 98–110, 2023.

[11] J. J. Liang, C. T. Yue, and B. Y. Qu, Multimodal multi-objective optimization: A preliminary study, in *Proc.* 2016 *IEEE Congress on Evolutionary Computation*, Vancouver, Canada, 2016, pp. 2454–2461.

[12] Q. Lin, W. Lin, Z. Zhu, M. Gong, J. Li, and C. A. C. Coello, Multimodal multiobjective evolutionary optimization with dual clustering in decision and objective spaces, *IEEE Trans. Evol. Comput.*, vol. 25, no. 1, pp. 130–144, 2021.

[13] Y. Liu, G. G. Yen, and D. Gong, A multimodal multiobjective evolutionary algorithm using two-archive and recombination strategies, *IEEE Trans. Evol. Comput.*, vol. 23, no. 4, pp. 660–674, 2019.

[14] W. Li, T. Zhang, R. Wang, and H. Ishibuchi, Weighted indicator-based evolutionary algorithm for multimodal multiobjective optimization, *IEEE Trans. Evol. Comput.*, vol. 25, no. 6, pp. 1064–1078, 2021.

[15] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, 2002.

[16] R. Tanabe and H. Ishibuchi, A framework to handle multimodal multiobjective optimization in decomposition-based evolutionary algorithms, *IEEE Trans. Evol. Comput.*, vol. 24, no. 4, pp. 720–734, 2020.

[17] Y. Peng and H. Ishibuchi, A diversity-enhanced subset selection framework for multimodal multiobjective optimization, *IEEE Trans. Evol. Comput.*, vol. 26, no. 5, pp. 886–900, 2022.

[18] Q. Fan and X. Yan, Solving multimodal multiobjective problems through zoning search, *IEEE Trans. Syst., Man, Cybern.: Syst.*, vol. 51, no. 8, pp. 4836–4847, 2021.

[19] G. Li, W. Wang, W. Zhang, W. You, F. Wu, and H. Tu, Handling multimodal multi-objective problems through self-organizing quantum-inspired particle swarm optimization, *Inf. Sci.*, vol. 577, pp. 510–540, 2021.

[20] Z. Li, J. Zou, S. Yang, and J. Zheng, A two-archive algorithm with decomposition and fitness allocation for multi-modal multi-objective optimization, *Inf. Sci.*, vol. 574, pp. 413–430, 2021.

[21] J. Liang, K. Qiao, C. Yue, K. Yu, B. Qu, R. Xu, Z. Li, and Y. Hu, A clustering-based differential evolution algorithm for solving multimodal multi-objective optimization problems, *Swarm Evol. Comput.*, vol. 60, p. 100788, 2021.

[22] B. Qu, G. Li, L. Yan, J. Liang, C. Yue, K. Yu, and O. D. Crisalle, A grid-guided particle swarm optimizer for multimodal multi-objective problems, *Appl. Soft Comput.*, vol. 117, p. 108381, 2022.

[23] K. Zhang, C. Shen, G. G. Yen, Z. Xu, and J. He, Two-stage double niched evolution strategy for multimodal multiobjective optimization, *IEEE Trans. Evol. Comput.*, vol. 25, no. 4, pp. 754–768, 2021.

[24] S. Han, K. Zhu, M. Zhou, and X. Cai, Information-utilization-method-assisted multimodal multiobjective optimization and application to credit card fraud detection, *IEEE Trans. Comput. Soc. Syst.*, vol. 8, no. 4, pp. 856–869, 2021.

[25] Y. Liu, H. Ishibuchi, G. G. Yen, Y. Nojima, and N. Masuyama, Handling imbalance between convergence and diversity in the decision space in evolutionary multimodal multiobjective optimization, *IEEE Trans. Evol. Comput.*, vol. 24, no. 3, pp. 551–565, 2020.

[26] C. Yue, P. N. Suganthan, J. Liang, B. Qu, K. Yu, Y. Zhu, and L. Yan, Differential evolution using improved crowding distance for multimodal multiobjective optimization, *Swarm Evol. Comput.*, vol. 62, p. 100849, 2021.

[27] Y. Tian, T. Zhang, J. Xiao, X. Zhang, and Y. Jin, A coevolutionary framework for constrained multiobjective optimization problems, *IEEE Trans. Evol. Comput.*, vol. 25, no. 1, pp. 102–116, 2021.

[28] J. Yuan, H. L. Liu, Y. S. Ong, and Z. He, Indicator-based evolutionary algorithm for solving constrained multiobjective optimization problems, *IEEE Trans. Evol. Comput.*, vol. 26, no. 2, pp. 379–391, 2022.

[29] Y. Tian, L. Si, X. Zhang, K. Tan, and Y. Jin, Local model-based Pareto front estimation for multiobjective optimization, *IEEE Trans. Syst., Man, Cybern.: Syst.*, vol.

53, no. 1, pp. 623–634, 2023.
[30] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [Educational Forum] , *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 73–87, 2017.
[31] R. B. Agrawal, K. Deb, and R. B. Agrawal, Simulated binary crossover for continuous search space, *Complex Syst.*, vol. 9, no. 3, pp. 115–148, 2000.
[32] H. Ishibuchi, R. Imada, N. Masuyama, and Y. Nojima, Comparison of hypervolume, IGD and IGD$^+$ from the viewpoint of optimal distributions of solutions, in *Proc. 10^{th} Int. Conf. Evolutionary Multi-Criterion Optimization*, East Lansing, MI, USA, 2019, pp. 332–345.
[33] Y. Tian, X. Xiang, X. Zhang, R. Cheng, and Y. Jin, Sampling reference points on the Pareto fronts of benchmark multi-objective optimization problems, in *Proc. 2018 IEEE Congress on Evolutionary Computation*, Rio de Janeiro, Brazil, 2018, pp. 1–6.
[34] A. Zhou, Q. Zhang, and Y. Jin, Approximating the set of Pareto-optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm, *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 1167–1189, 2009.
[35] H. Ishibuchi, L. M. Pang, and K. Shang, Difficulties in fair performance comparison of multi-objective evolutionary algorithms [research frontier] , *IEEE Comput. Intell. Mag.*, vol. 17, no. 1, pp. 86–101, 2022.
[36] J. Alcalá-Fdez, L. Sánchez, S. García, M. J. del Jesus, S. Ventura, J. M. Garrell, J. Otero, C. Romero, J. Bacardit, V. M. Rivas, et al., KEEL: A software tool to assess evolutionary algorithms for data mining problems, *Soft Comput.*, vol. 13, no. 3, pp. 307–318, 2009.
[37] E. Jiang, L. Wang, and J. Wang, Decomposition-based multi-objective optimization for energy-aware distributed hybrid flow shop scheduling with multiprocessor tasks, *Tsinghua Science and Technology*, vol. 26, no. 5, pp. 646–663, 2021.
[38] W. Zhang, X. Chen, and J. Jiang, A multi-objective optimization method of initial virtual machine fault-tolerant placement for star topological data centers of cloud systems, *Tsinghua Science and Technology*, vol. 26, no. 1, pp. 95–111, 2021.
[39] F. Zhao, X. Hu, L. Wang, and Z. Li, A memetic discrete differential evolution algorithm for the distributed permutation flow shop scheduling problem, *Complex Intell. Syst.*, vol. 8, no. 1, pp. 141–161, 2022.
[40] K. Gao, Y. Huang, A. Sadollah, and L. Wang, A review of energy-efficient scheduling in intelligent production systems, *Complex Intell. Syst.*, vol. 6, no. 2, pp. 237–249, 2020.
[41] H. Ma, H. Wei, Y. Tian, R. Cheng, and X. Zhang, A multi-stage evolutionary algorithm for multi-objective optimization with complex constraints, *Inf. Sci.*, vol. 560, pp. 68–91, 2021.
[42] J. Liang, X. Ban, K. Yu, B. Qu, K. Qiao, C. T. Yue, K. Chen, and K. C. Tan, A survey on evolutionary constrained multiobjective optimization, *IEEE Trans. Evol. Comput.*, vol. 27, no. 2, pp. 201–221, 2023.
[43] H. Ishibuchi, Y. Peng, and K. Shang, A scalable multimodal multiobjective test problem, in *Proc. 2019 IEEE Congress on Evolutionary Computation*, Wellington, New Zealand, 2019, pp. 310–317.
[44] Z. Cui, J. Zhang, Y. Wang, Y. Cao, X. Cai, W. Zhang, and J. Chen, A pigeon-inspired optimization algorithm for many-objective optimization problems, *Sci. China Inf. Sci.*, vol. 62, no. 7, p. 70212, 2019.
[45] R. Cheng, M. Li, Y. Tian, X. Zhang, S. Yang, Y. Jin, and X. Yao, A benchmark test suite for evolutionary many-objective optimization, *Complex Intell. Syst.*, vol. 3, no. 1, pp. 67–81, 2017.

Wenyin Gong received the BEng, MEng, and PhD degrees in computer science from China University of Geosciences, Wuhan, China, in 2004, 2007, and 2010, respectively. He is currently a professor with School of Computer Science, China University of Geosciences, Wuhan, China. His research interests include evolutionary algorithms, evolutionary optimization, and their applications. He has published over 80 research papers in journals and international conferences.

Fei Ming received the BS degree from China University of Geosciences, Wuhan, China, in 2019. He is currently pursuing the PhD degree with School of Computer Science, China University of Geosciences, Wuhan, China. His current research interests include evolutionary multiobjective optimization methods and their application.